Middlesex University London

Middlesex University Research Repository

An open access repository of

Middlesex University research

http://eprints.mdx.ac.uk

Wang, Chundong, Zhao, Zhentang, Gong, Liangyi, Zhu, Likun, Liu, Zheli and Cheng, Xiaochun ORCID logoORCID: https://orcid.org/0000-0003-0371-9646 (2018) A distributed anomaly detection system for in-vehicle network using HTM. IEEE Access, 6 . pp. 9091-9098. ISSN 2169-3536 [Article] (doi:10.1109/access.2018.2799210)

Published version (with publisher's formatting)

This version is available at: https://eprints.mdx.ac.uk/24573/

Copyright:

Middlesex University Research Repository makes the University's research available electronically.

Copyright and moral rights to this work are retained by the author and/or other copyright owners unless otherwise stated. The work is supplied on the understanding that any use for commercial gain is strictly forbidden. A copy may be downloaded for personal, non-commercial, research or study without prior permission and without charge.

Works, including theses and research projects, may not be reproduced in any format or medium, or extensive quotations taken from them, or their content changed in any way, without first obtaining permission in writing from the copyright holder(s). They may not be sold or exploited commercially in any format or medium without the prior written permission of the copyright holder(s).

Full bibliographic details must be given when referring to, or quoting from full items including the author's name, the title of the work, publication details where relevant (place, publisher, date), pagination, and for theses or dissertations the awarding institution, the degree type awarded, and the date of the award.

If you believe that any material held in the repository infringes copyright law, please contact the Repository Team at Middlesex University via the following email address:

eprints@mdx.ac.uk

The item will be removed from the repository while any claim is being investigated.

See also repository copyright: re-use policy: http://eprints.mdx.ac.uk/policies.html#copy

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 1, 2017, accepted January 3, 2018, date of publication January 30, 2018, date of current version March 13, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2799210

A Distributed Anomaly Detection System for In-Vehicle Network Using HTM

CHUNDONG WANG[1], ZHENTANG ZHAO[1], LIANGYI GONG[1], LIKUN ZHU[1], ZHELI LIU[2], AND XIAOCHUN CHENG[3], (Senior Member, IEEE)

[1]Key Laboratory of Computer Vision and System, Ministry of Education, Tianjin University of Technology, Tianjin 300384, China
[2]School of Computer Science and Engineering, Nankai University, Tianjin 300350, China
[3]Department of Computer Science, Middlesex University, London NW4 4BT, U.K.

Corresponding author: Liangyi Gong (gongliangyi@tjut.edu.cn)

This work was supported in part by the Foundation of the Educational Commission of Tianjin, China, under Grant 20130801, in part by the General Project of Tianjin Municipal Science and Technology Commission under Grant 15JCYBJC15600, in part by the Major Project of Tianjin Municipal Science and Technology Commission under Grant 15ZXDSGX00030, and in part by the NSFC: The United Foundation of General Technology and Fundamental Research under Grant U1536122.

ABSTRACT With the development of 5G and Internet of Vehicles technology, the possibility of remote wireless attack on an in-vehicle network has been proven by security researchers. Anomaly detection technology can effectively alleviate the security threat, as the first line of security defense. Based on this, this paper proposes a distributed anomaly detection system using hierarchical temporal memory (HTM) to enhance the security of a vehicular controller area network bus. The HTM model can predict the flow data in real time, which depends on the state of the previous learning. In addition, we improved the abnormal score mechanism to evaluate the prediction. We manually synthesized field modification and replay attack in data field. Compared with recurrent neural networks and hidden Markov model detection models, the results show that the distributed anomaly detection system based on HTM networks achieves better performance in the area under receiver operating characteristic curve score, precision, and recall.

INDEX TERMS In-vehicle network security, real-time anomaly detection, HTM algorithm.

I. INTRODUCTION

With the rapid development of mobile Internet, big data and cloud computing technology, the automobile gradually become intelligent, network oriented. The intelligent car network is a new direction of innovation and development. Concepts and technologies such as autopilot, shared car, and Internet of Vehicles (IoV) emerge as the times require. 5G is regarded by the industry as the key technology to realize automatic driving and network communication [1]. It has advantages of low delay, large bandwidth and high connection density. Moreover, most modern cars are equipped with multi-function remote information processing system, supporting global positioning system (GPS), media entertainment, or even directly accessing cellular networks. However, the remote information processing system is vulnerable to network attacks because it is connected to the external wireless network [2]–[4]. The attacker can access the target vehicle network through the wireless access interface [5], implement a variety of attacks such as replay attack, DoS attack, frame sniffing, frame injection and so on [6]–[8]. Thus, it brings potential security threats to the vehicle network.

The vehicle network is composed of Electronic Control Units (ECUs) and Controller Area Network (CAN) bus. And ECU is an embedded control component that connects sensors and actuators. Each ECU gets the input from its sensor to execute specific instructions through the executor, aiming to monitor the state of the vehicle and perform the corresponding behavior. Different ECUs can communicate with each other through the CAN bus. Even if they are deployed on a different speed bus, the architecture of the CAN network enables ECU to communicate with other ECU. When an attacker has a wireless remote access to the vehicle network, it can eavesdrop the communication between ECUs and send malicious control messages [9]–[11]. Therefore, we must improve the security capability of the vehicle network.

In order to strengthen the security of the car network, there are two main solutions. One is to design passive defense based on security protocol [12]–[14] or security network framework [15], [16], and the other is to

2169-3536 © 2018 IEEE. Translations and content mining are permitted for academic research only.
Personal use is also permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

detect potential network risk based on anomaly detection technology [17], [18]. Considering the security vulnerabilities existing in the CAN protocol, it plays a protective role that developing new security protocols or security frameworks for future cars. On the other hand, the researchers also put forward a variety of anomaly detection schemes, which set up and train detection models to discover network messages and alarm. However, most of the models have a single detection form, low reliability and non-real time detection. Therefore, our goal is to establish a more reliable and perfect vehicle network anomaly detection system.

The paper describes a distributed real-time anomaly detection system based on hierarchical temporal memory (HTM) learning algorithm [19], which shows better detection performance [20]. The HTM network provides a more accurate framework for the prediction, classification and anomaly detection [21]. Furthermore, the HTM network can learn the time based data sequence in a continuous online manner. Through our previous research work [22], the HTM prediction model can learn the data sequence of the vehicle network online. Finally, our system can learn CAN data online and detect abnormal states to generate alerts. We have made the following contributions:

1) A new distributed vehicle network anomaly detection framework has been proposed. The binary data stream before decoding is input into every data sequence predictor, and then the output prediction value is processed by the abnormal score mechanism. If abnormal condition occurs, the alarm signal is sent out. The detection device can be deployed directly on the CAN bus.
2) The CAN data sequence prediction based on HTM network is proposed. The HTM network has the characteristics of continuous online learning. When the input stream data changes, the memory of the model will be updated. And the detection system will also continue to learn new patterns in the CAN network when the firmware is upgraded or replaced.
3) A perfect exception scoring mechanism is proposed, which is used for abnormal decision making. The error measure is made using the predicted output value of the log loss function, and then the scores for a single data sequence can be derived. Finally, the appropriate overall rating type and threshold are chosen based on different IDs.

The rest of the paper is organized as follows. In Section 2, we introduce different approaches and techniques about research of in-vehicle networks security. In Section 3, we discuss some model and assumption. we specifically introduce our anomaly detection system for in-vehicle networks in Section 4. Further, we describe the results of the experiment and performance analysis in Section 5. Finally, we conclude the paper in Section 6.

II. RELATION WORK

Anomaly detection technology research has been widely carried out to help resisting malicious attacks of networks [23]–[25]. As a broadcast bus, the CAN bus does not specify which ECU generates CAN message, and malicious message also is used if included. Therefore, many studies choose intrusion detection mechanism to identify the source of the message without increasing the overhead of the system.

Taylor *et al.* [26] used a frequency-based anomaly detection mechanism. He transformed the field of industrial control into the CAN bus and proposed an algorithm to measure the insertion of anomalous frequencies between different groups on a sliding window. Compared with historical values, it generated an abnormal monitoring signal. At the same time, the information was inserted into the information training anomaly support vector machine (OCSVM) [27] for classified learning. The results showed that using the same information OCSVM could detect very short packet frequency insertion. The false alarm rate of this result was acceptable, but it still needs to be improved.

Nair *et al.* [28] set up OBD-SecureAlert, which was a warning mechanism to detect malicious packets of vehicles. It used the CAN message dataset when the car is attacked under the normal condition, which was used to generate the transition probability and the emission probability. They matched these probabilities to Hidden Markov Models (HMMs) [29] and analyzed the time series data to generate a test model. Then an alarm was issued when the monitor message appears abnormal status. This system can make an accurate judgment when one of its speed or driving speed changes or both change at the same time. However, there is no systematic warning given when multiple data changes simultaneously.

There are many machine learning technologies that are widely used in smart car intrusion detection technology. Chen *et al.* [30] proposed a frequency-based coding method for grouping features in artificial neural networks (ANN) [31], [32] and SVM [33], but its approach uses supervised techniques, so multiple marked datasets are needed in training. Compared with others method, Kang and Kang [34] adopted an unsupervised scheme and proposed an efficient intrusion detection system (IDS) based on Deep Neural Networks (DNN). First, they extract eigenvectors from in-vehicle network packet data and then train DNN parameters using both the pre-training method of Deep Belief Networks (DBN) and the traditional stochastic gradient descent method [35]. Taylor *et al.* [36] proposed an exception detector based on Recurrent Neural Networks (RNN) to detect CAN bus attacks. The detector works by learning and predicting the next data word from each transmitter on the bus. However, this method only detects a single ID exception and can not continue to learn online. And it does not compare with other anomaly detection methods.

This paper does not take the encryption authentication [37]–[39] method, because most of the certification mechanisms [40], [41] will increase the bus load and some mechanisms need to update the firmware or hardware. Although many researches can effectively warn CAN bus

TABLE 1. Data frame format of CAN 2.0B.

SOF field	ID field	Control field	Data field	CRC field	ACK field	EOF field
1 bit	11 bit	6 bit	0-64 bit	16 bit	1 bit	1 bit

invasion, these models have many shortcomings. Most are only for a single variable for intrusion detection. When multiple variables change at the same time, the detection rate of abnormal CAN message is very low. At the same time, most studies only detect threats that have already been detected. Once an intrusion message changes, the system can't effectively defend it.

III. CAN DATA STREAM AND ANOMALY STATE

A. CAN BUS DATA STREAM

In the CAN network, each ECU uses a data frame to transfer information to other ECUs on the CAN bus. The CAN bus is a broadcast medium. Without any routing mechanism, it is the responsibility of the receiver to determine which message is of interest. Most CAN bus hardware interfaces offer filtering capabilities to limit the packets its owner receives. The classical data frame format of CAN 2.0B packet is shown in Table 1. The arbitration field is consist of an 11 bit ID field, which is extended to 29 bits later with increase of network nodes. The data field includes a maximum of 8 bytes, which contains information to be transmitted between each other in ECU. Other fields are not introduced because it is not involved in the paper.

TABLE 2. CAN data analysis on impreza.

ID	Period	N_{bits}	$N_{packets}$	$\%_{unique}$
0D3	0.02	27	4355348	0.79
0D0	0.02	63	4354857	34.52
141	0.01	53	7995433	58.31
154	1.0	5	89522	0.017
282	0.05	27	1713455	17.43
360	0.05	32	1713657	68.91
370	0.05	45	1713792	4.63
374	1.0	6	84663	0.02
002	0.01	30	8552694	0.892
660	0.5	30	172455	100.00

We turn our attention to the data fields payloads of packets. Almost every ID contains 64-bit data fields. Security researchers and car hacking hobbyists typically resort to reverse-engineering the protocols by observing relationships between traffic and the vehicle's parameters and behaviour. However, it require difficult analysis skills and spend lots of time. So, we analyze data features directly extracted from a bitstream in the vehicular network before decoding. At the observed packet rates on the Impreza high speed bus, there is enough space in 64 bits to produce a novel data filed for each packet for the entire lifetime of the car. Table 2 shows the number and percentage of unique data fields payloads of packets observed over all the Impreza data captures for each ID. Considering that these data contain sensor readings, we expect it to be noisy and highly variable. However only one ID produces a novel word for every packet, and most are well below the full possible rate. The table also reports periodic with a fixed frequency and how many bits are used in 64-bit data fields for each ID. Even though most IDs transmit a full 64 bits, many of these bits are found to be constant over the entire data set.

B. CAN BUS ANOMALY STATE DEFINITION

On the CAN bus, the raw CAN bus data consists of a list of (ID, data payload) pairs indexed with timestamps. The sequence of packet data payloads associated with a given ID can be viewed as a multivariate binary sequence. Generally speaking, streaming data in CAN network is highly regular. Thus attacks usually include three performances: packets are added, packets are missing, or modified packets within a single ID's symbol stream. Effects on CAN bus traffic includes two major types of anomalies: frequency effects, and data effects. Frequency effects are in the context of all known IDs being periodic with fixed frequencies. They are defined as insertions of extra packets, or the erasure of expected packets. Data effects describe how data values can be changed in attack traffic. Advanced attacks usually have a malicious effect by setting control values in the data fields. Here we are concerned with pure data that does not involve additional packets.

All data field attacks can be described by the following parameters. Given an ID, a field within that ID's data protocol, for a duration d seconds that field is changed in one of the following ways:

Modification: the field is set to a constant value, such as the maximum or minimum possible value, or to an arbitrary constant.

Replay: the field is replaced with data from the same field captured at a different time. Only the data field is replaced. The rest of the data payload is left unchanged.

IV. DISTRIBUTED ANOMALY DETECTION SYSTEM USING HTM ALGORITHM

A. OVERVIEW OF PROPOSED ANOMALY DETECTION SYSTEM

The overall goal is to detect attacks in a car when they happen, while it is being driven. More precisely, the problem is detecting anomalous sequences on the CAN bus while the car is being driven. We assume we have a large corpus of known normal data to train the detectors, so our problem is a semi-supervised anomaly detection problem. One of the key challenges is distinguishing the natural novelty in the data from anomalies corresponding to attacks. And another key factor is that a production automotive anomaly detector must

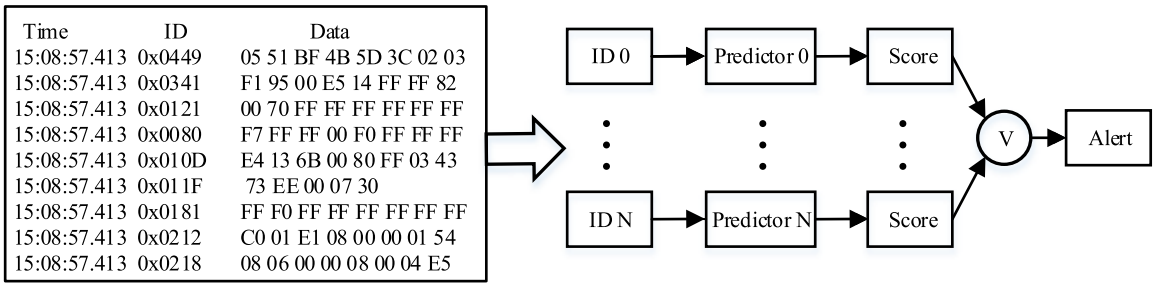


FIGURE 1. A Structure of the distributed anomaly detection system.

also work online, so that anomalies can be detected as soon as they manifest.

The structure of the distributed anomaly detection system is illustrated in Figure 1. Firstly, the data detection models directly process the sequence of packet data bits from a single ID. The input at each detector in the sequence are the bits from the packet's data field. Then the model using HTM algorithm learns to predict the next data fields based on training data. Moreover, after each bit score in the data fields of a packet is developed, it must be combined into a single score by loss function. Finally, we get an overall score within a time window for the full input sequence of a single ID. Our system will generate an alert, if the anomaly score of a single ID is going below a set threshold value.

B. PREDICTOR BASED ON HTM LEARNING ALGORITHM

HTM learning algorithm is a machine learning algorithm aimed at capturing the structure and algorithmic features of the new cerebral cortex. Biological studies have pointed out that not every cognitive function corresponds to a neural algorithm. As the loop of the new cerebral cortex is very uniform, the new cerebral cortex uses a set of common algorithms to perform different functions. The algorithm is based on flow data, not static data. It can be learned, identified and predicted by the latest input. It is a memory based system, the HTM network is trained by a large number of time-characteristic data and depends on a large number of pattern sequences. The core of HTM is the hierarchical organization structure, the construction of the region, the information based on time and the storage of data, which should be stored in discrete sparse representation. A HTM network is composed of layers arranged in a hierarchical area. This area is composed of a storage and prediction unit. And the multiple sub-Elements in the hierarchy are aggregated to form a storage unit. Because of the existence of the feedback connection, the information will also be diverted with the decline of the level. The implementation details of HTM learning algorithm are described in [42] and [43]. This paper use standard HTM system [44] and standard parameters to predict CAN data flow.

The data prediction model directly deals with the bit sequences from a single ID data domain. The input of each step in the sequence is the bit from the data field of the packet. So the input of the predictor model is a binary matrix (group number) × (inconstant bit number). In a production system, it is necessary to design a method to send data in a continuous or discontinuous window into a predictor.

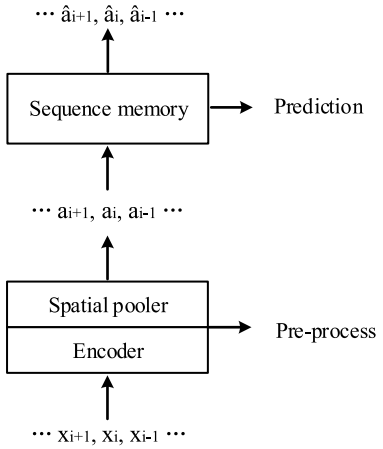


FIGURE 2. Proposed anomaly predictor based on HTM networks.

The anomaly detection problem is specifically focused on predicting next status according to the current previous states. To predict next state, the proposed anomaly predictor is shown in Figure 2. The input data is first processed using a component that includes an encoder and a sparse spatial pooling. With this component, the hidden features of the data can be more easily extracted. This method is more easily applied to high dimensional sequences, such as the CAN bus data sequence. The data after pretreatment is a sparse vector representing the input data. Then the sequence memory component is used to predict the convective data. This component models outputs a binary vector for a sparse vector of the current input data.

The predicting function is implemented based on HTM networks. We can predict current situation which is dependent on continuously learning previous states in CAN network. The prediction of the incoming data stream is directly related to the current packet detected and the current location of the packet within overall streaming data. The performance of the predictor is dependent on modeling the data in the sequence memory component. However, we can not directly determine whether there is any abnormality by prediction using HTM

networks. It needs to be evaluated by the post processing mechanism.

C. ANOMALY SCORE FOR DETECTION

For each bit of the input terminal, the logarithmic loss of each bit is the basis for the detection of abnormal signals. The detector model will output the prediction of the corresponding bit in the next word. Then, the logarithmic loss function can predict each data filed, which is between 0 and 1. This can be interpreted as the probability of the bit value. The log loss for current input is defined as:

$$L(\hat{b}_k,\ b_k) = -\Big(b_k \log(\hat{b}_k + \epsilon) + (1 - b_k)\log(1 - \hat{b}_k + \epsilon)\Big)$$

where b_k is k^{th} bit in a sparse vector representing the input data. $\hat{b}_k$ is k^{th} bit's predicted value using the sequence memory component based on previous sparse vector, and ϵ is a constant value that corrects the maximum loss.

The vehicle network design is made up of a specific module by multiple ECU interactions in order to achieve specific functions, such as the module includes the engine control module, the body control module, the instrument monitor display module and the remote information processing module. In addition, many behaviors require cooperation between different modules in order to complete some complex functions. For example, it requires complex interaction between the engine control module, the anti lock brake system, and the dashboard group to complete the automobile brake. For our detector to be practical, a single scalar value must be derived from all of the bit losses for all the data fields and IDs within a time window to alert for the entire system. Therefore, if we want to evaluate all the sequence data in anomaly detectors which operate on the CAN bus for all ID, we must get an overall score for input ID data within a time window.

It is challenging to merge the output of each exception detection model into a single decision. For a single ID, bit errors may vary across the data fields. The variance may be even greater for different IDs. Thus, our approach to combining scores is to define a serious of post-processing steps that convert a single ID's bit scores over time into a single output score. Two common combinations are used: maximum or window averaging. The average value method of the window calculates the average fraction of the sliding 0.1s time window, or directly returns the maximum anomaly score. For evaluating data anomaly scores for a single ID, we can choose the maximum of them all, or take individual thresholds and alarm for the entire system when any individual ID alarm exceeds a preset value.

Finally, the threshold must be selected for the entire detection model. A good way to choose the final decision threshold is to evaluate a model with a test data and select a threshold that can produce acceptable precision and recall balance.

The difference in the range of bytes is in the data domain, which will have a greater impact on the selection of the decision thresholds. Therefore, the different ID should adjust the single score threshold and appropriate combination score type should be select to improve the performance of the abnormal detector.

V. EXPERIMENT AND RESULTS

A. EXPERIMENT SETUP

We captured nearly 20 hours data from high CAN bus of Impreza. The data in its original state is a collection of text files containing comma separated values with a timestamp, ID, and data fields. Each file corresponds to a single drive, in most cases about 20 minutes long. We divided these files into different sets for training, validation, and testing. We assign 70% of the files to the training models and 10% files to the validation to avoid any overfitting in the training. The final 20% files was split into normal and simulated anomaly among the testing packets. Modification types of abnormal test packet include: minimum, maximum, constant (random), random and replay, as described above. In order to evaluate the impact of different detection time windows on the overall performance, the detection time windows are set to 0.2, 0.5, 1.0, 1.5 and 2 seconds respectively. Each ID's packet is a relatively independent data stream. And we train their respective detectors based on HTM networks for each ID. The performance of the HTM model is compared with that of the RNN and HMM models.

B. PERFORMANCE MEASURE

Precision and recall are the two basic evaluation indexes to measure the reliability of the anomaly detection system. Precision is defined as ratio of marked anomaly streaming data that is a true anomaly. Recall is the proportion of ratio of identified anomaly CAN data to actual anomaly data. The value of recall close to 1.0 represents the better performance of the detection.

We can measure different detection models by Receiver Operating Characteristic (ROC) and Area Under roc Curve (AUC). ROC is a diagram of the relationship between True Positive Rate (TPR) and False Positive Rate (FPR). We set a threshold for the two value classification problem and divide the instance into a positive class or a negative class (For example, the class that is greater than the threshold is a positive class). So we can change thresholds and classify them according to different thresholds. The area under the ROC curve is called AUC, and it can measure the performance of the classifier by a fixed value. In general, the value of AUC is between 0.5 and 1, and the larger AUC represents a better Performance.

Due to the use of different combination scoring methods and different thresholds, different ID anomaly detection will produce greater differences. So we first evaluated this effect, as is shown in table 3. The table includes the test data precision, and recall for the different individual IDs on the post processing steps. nce setting different thresholds, precision and score for each ID can reach more than 98%. But what we need to do is to improve the recall rate, when the accuracy is high.

TABLE 3. Post processing methods for the HTM model.

ID	Score type	Threshold	Precision	Recall
0D0	average	2.432	0.9912	0.263
0D1	max	13.755	0.9894	0.712
141	max	7.562	0.9941	0.476
218	average	2.242	0.9908	0.692
282	average	1.675	0.9976	0.501
340	average	2.114	0.9839	0.564
360	max	2.432	0.9834	0.254
370	average	2.553	0.9862	0.596

C. FIELD MODIFICATION RESULTS

We tested the performance of all the models with the score types and decision thresholds defined above. The size of the window can affect the performance of the model, so we combine the flow model and window of different lengths as a separate method to evaluate it. Longer windows produce more reliable statistics, but shorter windows produce faster results. Therefore, it is worth exploring the relationship between window length and performance. We evaluated window length of 0.2, 0.5 and 1 seconds when the field is modified.

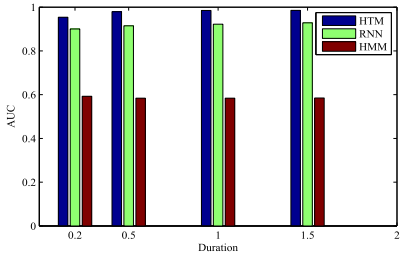


FIGURE 3. AUC score for data sequence field modification case.

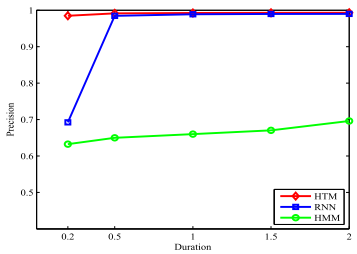


FIGURE 4. Precision on field modification replacement test case.

The values of AUC scores is given in Figure 3. The HTM model is the best. It is followed by the RNN and HMM networks are smallest. With the change of the anomaly detection window, the accuracy and recall of various models are shown in Figure 4 and Figure 5 respectively. It is worth noting that the detection accuracy of the HTM model is much higher than that of the RNN and HMM models when the detection window time is set to 0.2 seconds. And the recall rate of the HMM model is the best. However, with the increase of the time length of the detection window, the accuracy and recall rate of RNN has been greatly improved, and it has been approaching the HTM model constantly, while the change of

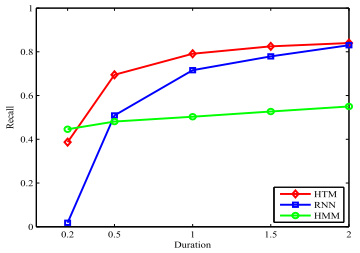


FIGURE 5. Recall on field modification replacement test case.

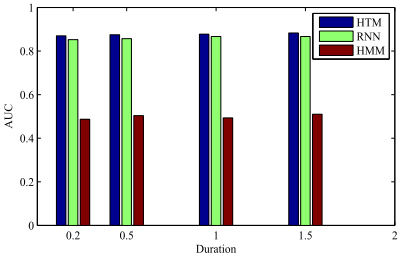


FIGURE 6. AUC score for data sequence replay case.

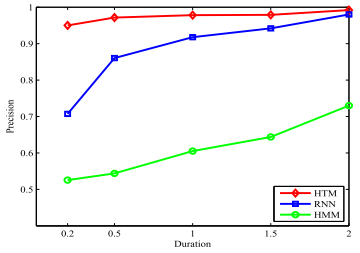


FIGURE 7. Precision on replay field replacement test case.

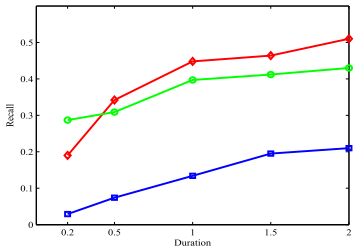


FIGURE 8. Recall on replay field replacement test case.

HMM is slower. Considering all the factors, the performance of the exception detection system based on HTM model is optimal when data field is modified.

D. FIELD REPLAY RESULTS

Similarly, we test all models of replay attacks. The values of AUC scores for the test replay case is given in Figure 6. Compared with other models, HTM still keeps the highest AUC score in all time windows. The accuracy and recall of the model are shown in Figure 7 and Figure 8 respectively. The recall rate of the HMM model is slightly lower than the HTM model and higher than the RNN model. The test results of the HTM model are still better in accuracy and recall. Compared with results of data field modification, RNN's rate

of precision growth slows down, with the extension of the time window and the recall efficiency of all models is not ideal. Therefore, the detection of data field replay is more difficult than data field modification. In fact, replaying data will keep the other domain features consistent so the data will not change much. This increases the difficulty of abnormal detection. All models should try to improve the recall rate when the accuracy rate is guaranteed.

VI. CONCLUSION

In this paper, a distributed anomaly detection system based on HTM learning algorithm is introduced, which is used to detect the data sequence anomaly of the vehicle CAN bus network. The detection method can monitor all ID exceptions at the same time, without professional knowledge about reverse and vehicular bus protocols. Because HTM network has the advantage of online learning for streaming data, the algorithm can detect not only the known type attacks of CAN bus, but also the anomaly detector can learn online from CAN data stream continuously, and detect unknown attacks. Through the overall evaluation of the system, it is proved that the system can get more reliable detection effect compared with other existing CAN data domain anomaly detection methods, such as hidden Markov model and RNN model. However, more work need to be done to improve the performance of the system in more complex situations. For example, the training time of the model can be reduced and the exception detection efficiency will be improved by removing the redundant fields in the data domain. In addition, we also have to integrate ID with data dependencies and improve the combination evaluation of multiple ID to achieve better overall detection performance, because the ID of each monitor increases the possibility of false alarm. This is to avoid the greater impact of a single special ID judgment on the whole system. At the same time, a real attack datas are needed to test the system.

Furthermore, if the detection system continuously monitors the vehicle network and finds an exception, it appears a new challenge of the real-time system response. Such a response may require an additional design of a separate component. This component can enable the vehicle to detect the response of the attack and immediately start the security mode. Thus the vehicle is allowed to be safely parked. In fact, the coordination of this kind of response may be more complex than improving the performance of the exception detection because it requires coordination between multiple components. Therefore, the design goal of automobile anomaly detection system must also include intelligent interaction and emergency handling mechanism.

Finally, a single intrusion detection module can not meet the security needs of automotive intelligent network. The vehicle network security system needs a more lightweight authentication encryption mechanism. The vehicle network architecture needs to be upgraded to a higher level of architecture. At the same time, the bus security protocol should be improved. All these aspects are worth further studying in the future.

REFERENCES

[1] C.-X. Wang *et al.*, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[2] C. Miller and C. Valasek, "Remote exploitation of an unaltered passenger vehicle," Black Hat USA, Aug. 2015. [Online]. Available: http://illmatics.com/Remote Car Hacking.pdf

[3] S. Woo, H. J. Jo, and D. H. Lee, "A practical wireless attack on the connected car and security protocol for in-vehicle CAN," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 2, pp. 993–1006, Apr. 2015.

[4] J. H. Kim, S.-H. Seo, N. T. Hai, B. M. Cheon, Y. S. Lee, and J. W. Jeon, "Gateway framework for in-vehicle networks based on CAN, FlexRay, and Ethernet," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4472–4486, Oct. 2015.

[5] K. Koscher *et al.*, "Experimental security analysis of a modern automobile," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2010, pp. 447–462.

[6] C. Miller and C. Valasek, "Adventures in automotive networks and control units," *Def Con*, vol. 21, pp. 260–264, Aug. 2013.

[7] C. Miller and C. Valasek, "A survey of remote automotive attack surfaces," in *Proc. Scribd*, vol. 21. Washington, DC, USA, 2014, pp. 34–90.

[8] H. J. Jo, W. Choi, S. Y. Na, S. Woo, and D. H. Lee, "Vulnerabilities of Android os-based telematics system," *Wireless Pers. Commun.*, vol. 92, no. 4, pp. 1511–1530, 2017.

[9] T. Hoppe, S. Kiltz, and J. Dittmann, "Security threats to automotive CAN networks—Practical examples and selected short-term countermeasures," *Rel. Eng. Syst. Safety*, vol. 96, no. 1, pp. 11–25, 2011.

[10] A. Y. Javaid, W. Sun, V. K. Devabhaktuni, and M. Alam, "Cyber security threat analysis and modeling of an unmanned aerial vehicle system," in *Proc. IEEE Conf. Technol. Homeland Secur. (HST)*, Nov. 2012, pp. 585–590.

[11] D. Ward, I. Ibarra, and A. Ruddle, "Threat analysis and risk assessment in automotive cyber security," *SAE Int. J. Passenger Cars-Electron. Elect. Syst.*, vol. 6, no. 2, pp. 507–513, 2013.

[12] C.-W. Lin and A. Sangiovanni-Vincentelli, "Cyber-security for the controller area network (CAN) communication protocol," in *Proc. Int. Conf. Cyber Secur. (CyberSecurity)*, Dec. 2012, pp. 1–7.

[13] B. Groza and S. Murvay, "Efficient protocols for secure broadcast in controller area networks," *IEEE Trans. Ind. Informat.*, vol. 9, no. 4, pp. 2034–2042, Nov. 2013.

[14] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee. (2016). "Identifying ECUs using inimitable characteristics of signals in controller area networks." [Online]. Available: https://arxiv.org/abs/1607.00497

[15] Q. Wang and S. Sawhney, "VeCure: A practical security framework to protect the CAN bus of vehicles," in *Proc. Int. Conf. IEEE Internet Things (IOT)*, Oct. 2014, pp. 13–18.

[16] P. Kleberger, N. Nowdehi, and T. Olovsson, "Towards designing secure in-vehicle network architectures using community detection algorithms," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Dec. 2014, pp. 69–76.

[17] M. Müter and N. Asaj, "Entropy-based anomaly detection for in-vehicle networks,"in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 1110–1115.

[18] H. M. Song, H. R. Kim, and H. K. Kim, "Intrusion detection system based on the analysis of time intervals of CAN messages for in-vehicle network," in *Proc. Int. Conf. IEEE Inf. Netw. (ICOIN)*, Jan. 2016, pp. 63–68.

[19] D. George and J. Hawkins, "A hierarchical Bayesian model of invariant pattern recognition in the visual cortex," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 3. Jul. 2005, pp. 1812–1817.

[20] D. E. Padilla, R. Brinkworth, and M. D. McDonnell, "Performance of a hierarchical temporal memory network in noisy sequence learning," in *Proc. IEEE Int. Conf. Comput. Intell. Cybern. (CYBERNETICSCOM)*, Dec. 2013, pp. 45–51.

[21] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," *Neurocomputing*, vol. 262, pp. 134–147, Nov. 2017.

[22] C. Wang *et al.*, "Unsupervised real-time anomaly detection system for vehicular network," *Res. Briefs Inf. Commun. Technol. Evol.*, vol. 3, no. 5, pp. 1–11, Oct. 2017.

[23] Z. Hui, "Application of anomaly detection technology in email virus prevention," in *Proc. Int. Conf. Electr. Inf. Control Eng. (ICEICE)*, Apr. 2011, pp. 4796–4797.

[24] M. Kiermeier, M. Werner, C. Linnhoff-Popien, H. Sauer, and J. Wieghardt, "Anomaly detection in self-organizing industrial systems using pathlets," in *Proc. IEEE Int. Conf. Ind. Technol. (ICIT)*, Mar. 2017, pp. 1226–1231.

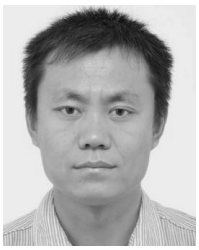
[25] J.-Q. Wei, Q.-L. Zhang, and X. Li, "Network anomaly detection and localization," in *Proc. 13th Int. Comput. Conf. Wavelet Active Media Technol. Inf. Process. (ICCWAMTIP)*, Dec. 2016, pp. 8–13.
[26] A. Taylor, N. Japkowicz, and S. Leblanc, "Frequency-based anomaly detection for the automotive CAN bus," in *Proc. World Congr. Ind. Control Syst. Secur.*, Dec. 2015, pp. 45–49.
[27] H. J. Shin, D.-H. Eom, and S.-S. Kim, "One-class support vector machines—An application in machine fault detection and classification," *Comput. Ind. Eng.*, vol. 48, no. 2, pp. 395–408, 2005.
[28] S. Nair, S. Mittal, and A. Joshi, "OBD SecureAlert: An anomaly detection system for vehicles," in *Proc. IEEE Int. Conf. Smart Comput.*, May 2016, pp. 1–6.
[29] S. R. Eddy, "Hidden Markov models," *Current Opinion Struct. Biol.*, vol. 6, no. 6, pp. 361–365, 1996.
[30] W.-H. Chen, S.-H. Hsu, and H.-P. Shen, "Application of SVM and ANN for intrusion detection," *Comput. Oper. Res.*, vol. 32, no. 10, pp. 2617–2634, 2005.
[31] V. Golovko and P. Kochurko, "Intrusion recognition using neural networks," in *Proc. Intell. Data Acquisition Adv. Comput. Syst., Technol. Appl. (IDAACS)*, Sep. 2005, pp. 108–111.
[32] Z. Zhang, J. Li, C. N. Manikopoulos, J. Jorgenson, and J. Ucles, "HIDE: A hierarchical network intrusion detection system using statistical preprocessing and neural network classification," in *Proc. IEEE Workshop Inf. Assurance Secur.*, Jun. 2001, pp. 85–90.
[33] W. Hu, Y. Liao, and V. R. Vemuri, "Robust anomaly detection using support vector machines," in *Proc. Int. Conf. Mach. Learn.*, 2003, pp. 282–289.
[34] M.-J. Kang and J.-W. Kang, "Intrusion detection system using deep neural network for in-vehicle network security," *PLoS ONE*, vol. 11, no. 6, p. e0155781, 2016.
[35] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Neurocomputing: Foundations of Research*. Cambridge, MA, USA: MIT Press, 1986, pp. 318–362.
[36] A. Taylor, S. Leblanc, and N. Japkowicz, "Anomaly detection in automobile control network data with long short-term memory networks," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2016, pp. 130–139.
[37] J. Li, Y. Zhang, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing," *Comput. Secur.*, vol. 72, pp. 1–12, Jan. 2018, doi: 10.1016/j.cose.2017.08.007.
[38] Z. Liu, T. Li, P. Li, C. Jia, and J. Li, "Verifiable searchable encryption with aggregate keys for data sharing system," *Future Generat. Comput. Syst.*, vol. 78, pp. 778–788, 2018. [Online]. Available: https://doi.org/10.1016/j.future.2017.02.024
[39] B. Cui, Z. Liu, and L. Wang, "Key-aggregate searchable encryption (KASE) for group data sharing via cloud storage," *IEEE Trans. Comput.*, vol. 65, no. 8, pp. 2374–2385, Aug. 2016.
[40] J. Li, X. Chen, M. Li, J. Li, P. P. C. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 6, pp. 1615–1625, Jun. 2014.
[41] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2201–2210, Aug. 2014.
[42] J. Hawkins and S. Ahmad, "Why neurons have thousands of synapses, a theory of sequence memory in neocortex," *Frontiers Neural Circuits*, vol. 10, no. 23, pp. 1–13, Mar. 2016
[43] S. Ahmad and J. Hawkins. (2015). "Properties of sparse distributed representations and their application to hierarchical temporal memory." [Online]. Available: https://arxiv.org/abs/1503.07469
[44] Y. Cui, S. Ahmad, and J. Hawkins, "Continuous online sequence learning with an unsupervised neural network model," *Neural Comput.*, vol. 28, no. 11, pp. 2474–2504, 2016.

CHUNDONG WANG received the B.Sc. degree in computer science from Tianjin Normal University, China, in 1991, and the M.Sc. and Ph.D. degrees in computer science from Nankai University, China, in 2002 and 2007, respectively. He is currently a Professor with the Tianjin University of Technology. His current research interests include network information security, pervasive computing, mobile computing, and intelligent information processing.

ZHENTANG ZHAO received the B.S. degree in electronic and information engineering from Liaocheng University in 2015. He is currently pursuing the master's degree in information and communication engineering with the Tianjin University of Technology, Tianjin, China. His research interests include wireless security and security of vehicle networking.

LIANGYI GONG received the B.E. and Ph.D. degrees from the Department of Computer Science and Technology, Harbin Engineering University, Harbin Engineering University, China, in 2010 and 2016, respectively. He is currently a Lecturer with the School of Computer Science and Engineering, Tianjin University of Technology. His major research interests cover mobile/pervasive computing, wireless sensor networking, and network information security.

LIKUN ZHU received the B.S. degree in communication engineering from the North China University of Science and Technology in 2016. He is currently pursuing the M.Sc. degree in information and communication engineering with the Tianjin University of Technology, China. His main research is directed to the architecture of wireless security, indoor locations, and wireless perception.



ZHELI LIU received the B.Sc. and M.Sc. degrees in computer science and the Ph.D. degree in computer applications from Jilin University, China, in 2002, 2005, and 2009, respectively. He was a Post-Doctoral Fellow with Nankai University. He joined the College of Computer and Control Engineering, Nankai University, in 2011, where he is currently an Associate Professor. His current research interests include applied cryptography and data privacy protection.

XIAOCHUN CHENG (SM'04) received the B.Eng. degree in computer software and the Ph.D. degree in artificial intelligence from Jilin University, China, in 1992 and 1996, respectively. He is a member of the IEEE SMC: Technical Committee on Systems Safety and Security. He is also a Committee Member of the European Systems Safety Society. He is the Secretary of the IEEE SMC, United Kingdom and Republic of Ireland.